WHAT IS CLAIMED IS:

1. A method for combining two data streams, comprising:

interpolating one or more samples between existing samples of one of the two data streams;

adjusting a number of samples of said one of the two data streams to maintain balance in a downstream synchronizing buffer.

2. The method according to claim 1, wherein said adjusting comprises adding or decimating samples from the interpolated samples.

3. The method according to claim 1, further comprising combining the one of said two data streams with the other of said two data streams after said adjusting of said one data stream.

4. The method according to claim 1, further comprising:

detecting a number of samples in the synchronizing buffer;

upon detecting a number of samples below a predetermined lower threshold, inputting a regular sample and last phase delayed sample into the synchronizing buffer and then inputting a predetermined number of phase delayed samples in reverse order into the synchronizing buffer; and

upon detecting a number of samples in the synchronizing buffer above a predetermined upper threshold, inputting a predetermined number of phase delayed

samples in order into the synchronizing buffer and then disabling writing into the synchronizing buffer for one cycle.

5. An apparatus for combining two data streams comprising:

a first buffer receiving a first data stream of the two data streams and having an input being clocked in by a first sample clock associated with the first data stream and having an output being clocked out by the first sample clock;

an interpolating filter receiving the second data stream and outputting a decimated oversampled version of the second data stream;

a multiplexer having a first input being coupled to an output of the interpolating filter, having a second input receiving the second data stream and outputting a modified data stream;

a second buffer receiving the modified data stream, having an input being clocked in by a second sample clock associated with the second data stream, having an output being clocked out by the first sample clock and having a level monitor output;

a buffer controller having an input being coupled to the level monitor output of the second buffer, having a first output controlling an output of the multiplexer, having a second output controlling the output of the interpolating polyphase filter and having a third output controlling the output of the second buffer.

6. The apparatus according to claim 5, wherein the buffer controller monitors a number of samples in the second buffer, and upon detecting a number of

samples below a predetermined lower threshold, enables inputting of a regular sample and a last phase delayed sample from the interpolating filter into the second buffer, and enables inputting a predetermined number of phase delayed samples in reverse order into the second buffer; and upon detecting a number of samples in the second buffer above a predetermined upper threshold, enables inputting a predetermined number of phase delayed samples in order into the second buffer and then disables writing into the second buffer for one cycle.

7. The apparatus according to claim 5, further comprising an adder having two inputs being coupled to the outputs of the first and second buffer and providing a combined data output.

8. The apparatus according to claim 5, wherein said buffer controller disables writing into the second buffer by the multiplexer when the buffer level increases by one sample.

9. The apparatus according to claim 5, wherein said buffer controller disables reading out of the second buffer when the buffer level decreases by one sample.

10. The apparatus according to claim 5, wherein said buffer controller causes the interpolating filter to decimate samples when the buffer level increases by one sample.

11. The apparatus according to claim 5, wherein said buffer controller causes the interpolating filter to add samples when the buffer level decreases by one sample.

12. The apparatus according to claim 5, wherein the interpolating filter comprises a plurality of phases (m) each outputting a delayed version the second data stream by successive increments of $360°/m$ and a multiplexer being coupled to the outputs of the plurality of phases and outputting the decimated oversampled version of the second data stream.

13. The apparatus according to claim 12, wherein said buffer controller controls the output of the multiplexer of the interpolating filter to decimate samples when the buffer level increases by one sample and to add samples when the buffer level decreases by one sample.

14. The apparatus according to claim 5, further comprising a memory and a switch, wherein the interpolating filter comprises a single finite impulse response filter having a plurality of coefficients, said memory storing a plurality (m) of sets of coefficients, one set for each of m phases, and said switch replaces the plurality coefficients for use by the finite impulse response filter as needed for each phase.

15. A method for combining two asynchronous data streams having clocks offset in frequency comprising:

clocking a first data stream into and out of a first buffer using a first clock associated with the first data stream;

clocking a second data stream into a second buffer using a second clock associated with the second data stream and clocking the second data stream out of the second buffer using the first clock;

interpolating and decimating samples of the second data stream prior to clocking the second data stream into the second buffer based on an overflow or underflow of the second buffer; and

combining the outputs of the first and second buffers.

16. The method according to claim 15, further comprising dropping a sample from the second buffer when a buffer level of the second buffer increases by one sample.

17. The method according to claim 15, further comprising adding a sample to the second buffer when a buffer level of the second buffer decreases by one sample.

18. The method according to claim 15, further comprising disabling writing into the second buffer when a buffer level of the second buffer increases by one sample.

19. The method according to claim 15, further comprising disabling reading out of the second buffer when a buffer level of the second buffer increases by one sample.

20. The method according to claim 15, wherein the step of interpolating and decimating further comprises delaying the second data stream by a plurality of phase delays in parallel and multiplexing the plurality of delays into a single stream.

21. The method according to claim 20, further comprising storing in memory a set of coefficients for each of the plurality of phase delays.

22. The method according to claim 21, further comprising using a single finite impulse response filter to perform the plurality of phase delays and switching between the sets of coefficients stored in memory as needed to provide each phase of the interpolating filter.

23. The method according to claim 15, wherein the step of interpolating and decimating comprises:

detecting a number of samples in the second buffer;

upon detecting a number of samples in the second buffer below a predetermined lower threshold, inputting a regular sample and last phase delayed

sample into the second buffer and then inputting a predetermined number of phase delayed samples in reverse order into the second buffer; and

upon detecting a number of samples in the second buffer above a predetermined upper threshold, inputting a predetermined number of phase delayed samples in order into the second buffer and then disabling writing into the second buffer for one cycle.